

# Updates of the WONDER software interface for using Wave Field Synthesis

Marije A.J. BAALMAN  
Communication Sciences, Technische Universität Berlin  
Sekt. EN8, Einsteinufer 17  
Berlin, Germany  
baalman@kgw.tu-berlin.de

## Abstract

WONDER is a software interface for using Wave Field Synthesis for audio spatialisation. Its user group is aimed to be composers of electronic music or sound artists. The program provides a graphical interface as well as the possibility to control it externally using the OpenSoundControl protocol. The paper describes improvements and updates to the program, since last year.

## Keywords

Wave Field Synthesis, spatialisation

## 1 Introduction

Wave Field Synthesis (WFS) is a technique for sound spatialisation, that overcomes the main shortcoming of other spatialisation techniques, as there is a large listening area and no “sweet spot”. In recent years, WFS has become usable with commercially available hardware.

This paper describes the further development of the WONDER program, that was designed to make the WFS-technique available and usable for composers of electronic music and sound artists.

## 2 Short overview of WFS and WONDER

WFS is based on the principle of Huygens, which states that a wave front can be considered as an infinite number of point sources, that each emit waves; their wavefronts will add up to the next wavefronts. With Wave Field Synthesis, by using a discrete, linear array of loudspeakers, one can synthesize correct wavefronts in the horizontal plane (Berkhout e.a. 1993). See figure 1 for an illustration of the technique.

WONDER is an open source software program to control a WFS system. The program provides a graphical user interface and allows the user to

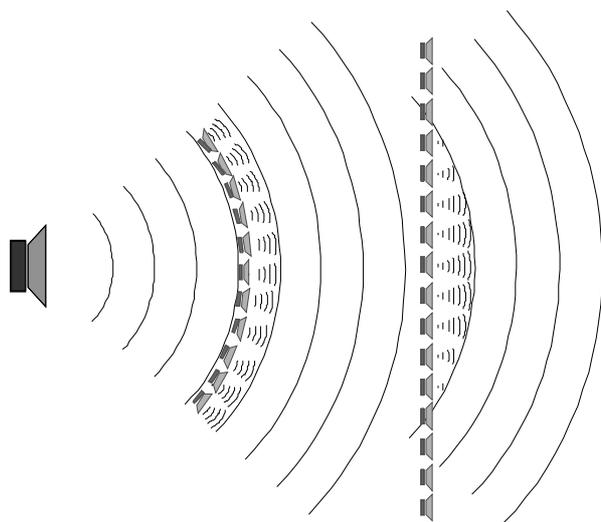


Figure 1. The Huygens' principle (left) and the Wave Field Synthesis principle (right).

think in terms of positions and movements, while the program takes care of the necessary calculations for the speaker driver functions. The program is built up in three parts: a grid definition tool, a composition tool and a play interface. Additional tools allow the user to manipulate grids or scores, or view filter data.

For the actual realtime convolution, WONDER relies on the program BruteFIR (Torgler). This program is capable of doing the amount of filter convolutions that are necessary to use WFS in realtime. On the other hand, BruteFIR has the drawback, that all the filters need to be calculated beforehand and during runtime need to be stored in RAM. This limits the flexibility for realtime use. It is for this reason, that a grid of points needs to be defined in advance, so that the filters for these points can be calculated beforehand and stored in memory.

For a more complete description of the WONDER software and the WFS-technique in general, I refer back to previous papers (Baalman, 2003/2004).

### 3 Updates to WONDER

Since its initial release in July 2004, WONDER has gone through some changes and updates. New tools have been implemented and the functionality of some of the old tools have been improved. Also, some parts were reimplemented to allow for easier extension in the future and resulting in a cleaner and more modular design.

The graphical overview, which displays the spatial layout of the speakers and source positions have been made consistent with each other and now all provide the same functionality, such as the view point (a stage view or an audience view), whether or not to show the room, setting the limits and displaying a background image.

The room definition has been improved: it is now possible to define an absorption factor for each wall, instead of one for all walls.

#### 3.1 Grid tools

The Grid definition tool allows a user to define a grid consisting of various segments. Each segment can have a different spacing of points within its specified area and different characteristics, such as inclusion of high frequency damping and room parameters for reflections.

The menu “Tools” now provides two extra tools to manipulate Grids. It is possible to merge several grids to one grid and to transform a grid.

The Merge-tool puts the points of several grids into one grid, allowing the user to use more than one grid in an environment.

The Transform-tool applies several spatial transformations to the segments of a grid and then calculates the points resulting from these transformed segments. This transformation can be useful if a piece will be performed on another WFS system, which has another geometry of the speaker setup and the coordinates of the grid need to be transformed.

The filter view (fig. 2) is a way to verify a grid graphically. It shows you the coefficients of all the filters of a source position in a plot. The plot shows on the horizontal axis the loudspeakers, in the vertical direction the time (on the top is zero). The intensity indicates the value of the absolute value of the volume. Above the graph, the input parameters of the gridpoint are given, as well as some parameters of the grid as a whole. This filter

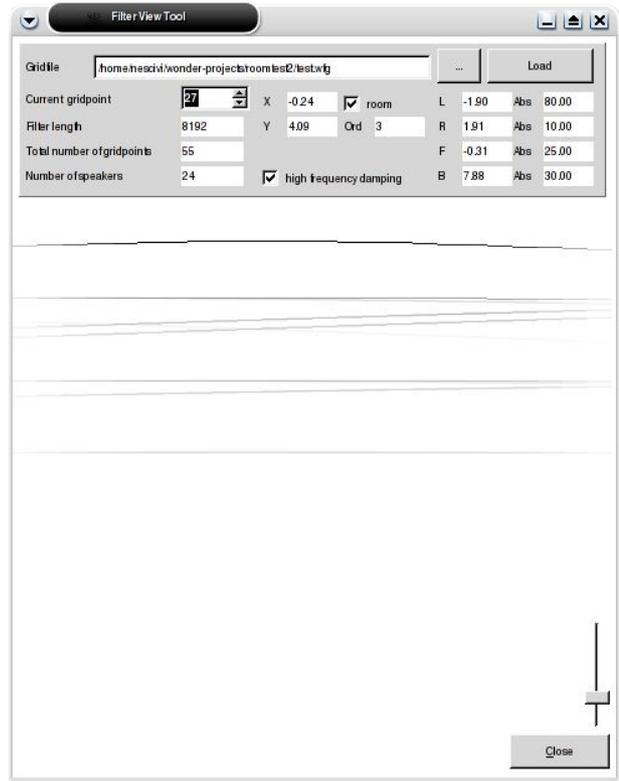


Figure 2. Filter view tool of WONDER. At the top, information about the grid and the current grid point are given. In the plot itself is in the horizontal direction the speakers, in the vertical direction the time. The intensity (contrast can be changed with the slider at the bottom right) is an indication of the strength of the pulse. The view clearly shows the reflection pattern of the impulse response.

overview can be useful for verification of calculations or for educational purposes.

Another way to verify a grid is using the grid test mode during playback with which you can step through the points of a grid and listen to each point separately.

#### 3.2 Composition tools

With the composition tool the user can define a spatial composition of the sound source movements. For each source the movement can be divided in sections in time and the spatial parameters can be given segmentwise per section.

In the composition definition dialog it is also possible to transform the current composition. The user can define a set of spatial transformations that have to be applied to the sources and sections specified. After the transformations have been applied, the user can continue working on the composition. This tool is especially handy when one source has to make a similar movement as

another: just make a copy of the one source and apply a transformation to the copy.

After a score has been created (either with the composition tool or by recording a score), there are four tools available in the “Tools”-menu to manipulate the scores.

“Clean score” is handy for a recorded score. This cleans up any double time information and rounds the times to the minimum time step (default: 25 ms).

“Merge scores” enables you to merge different scores into one. It allows a remapping of sources per score included.

“Transform score” allows you to make transformations to different sources in a score.

The last tool is the “timeline view”, which shows the x- and y-component in a timeline (fig. 4); it shows the selected time section as a path in an x-y view. It is also possible to manipulate time points in this view. While playing it shows a playhead to indicate the current time. The concept of this timeline view is inspired by the program “Meloncillo” (Rutz). The timeline view allows for a different way of working on a composition: the user can directly manipulate the score.

### 3.3 Play interface

WONDER provides a graphical interface to move sound sources or to view the movement of the sound sources. The movement of sound



Figure 3. Transport control of WONDER. The slider enables the user to jump to a new time. The time in green (left) indicates the running time, the time in yellow (right) the total duration of the score. The caption of the window indicates the name of the score file.

sources can be controlled externally with the OSC-protocol (Wright, 2003).

Score control is possible by using the Transport controls (fig. 3), which have been re-implemented.

The sound in- and output can be chosen to be OSS, ALSA, JACK or a sound file. In the first three cases the input has to be equal to the output.

The program BruteFIR (Torger) is used as audio engine. The communication between BruteFIR and WONDER can be verified by using a logview, which displays the output of BruteFIR. Due to some changes in the command line interface of BruteFIR, the communication between WONDER and BruteFIR could be improved and there is no more a problem with writing and reading permissions for the local socket.

The logview, which records (in red) the messages that are shown in the statusbar of WONDER, shows the feedback from the play engine BruteFIR in black. It is possible to save the log to a file.

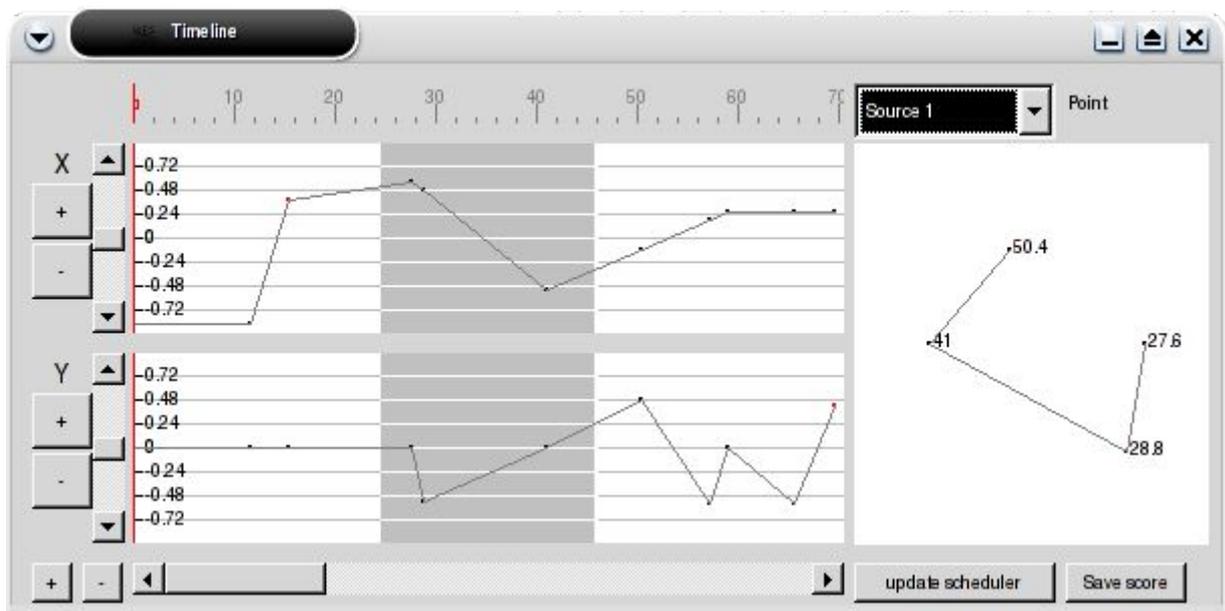


Figure 4. The timeline view of WONDER. The selected time section is shown as a path in the positional overview. The user can edit the score by moving the breakpoints or adding new ones.

### 3.4 Help functions

The manual of the program is accessible via the “Contents” item in the “Help”-menu. This displays a simple HTML-browser with which you can browse through the documentation. Alternately, you can use your own favourite browser to view the manual.

Additionally, in most views of WONDER, hovering above buttons, gives you a short explanation about the buttons' functionality.

### 4 External programs to control WONDER

There are two examples available which show how to control WONDER from another program: a SuperCollider Class available and an example MAX/MSP patch.

Another program that can be used for external control is the JAVA-program SpaceJockey (Kneppers), which was designed to enable the use of (customizable) movement patterns and to provide MIDI-control over the movements.

### 5 Conclusion

WONDER has improved in the last year and has become more stable and more usable. Several changes have been made to facilitate further development of the program.

Current work is to create an interface to EASE for more complex room simulation and to integrate the use of SuperCollider as an optional engine for the spatialisation. It is expected that SuperCollider can provide more flexibility, such as removing the necessity to load all filter files in RAM and the possibility to calculate filter coefficients during runtime.

Current research is focused on how to implement a more complex sound source definition.

### Download

A download is available at:

<http://gigant.kgw.tu-berlin.de/~baalman/>

### References

- Baalman, M.A.J., 2003, *Application of Wave Field Synthesis in the composition of electronic music*, International Computer Music Conference 2003, Singapore
- Baalman, M.A.J., 2004, *Application of Wave Field Synthesis in electronic music and sound installations*, Linux Audio Conference 2004, ZKM, Karlsruhe, Germany
- Baalman, M.A.J. & Plewe, D., 2004, *WONDER - a software interface for the application of Wave Field Synthesis in electronic music and interactive sound installations*, International Computer Music Conference 2004, Miami, Fl., USA
- Berkhout, A.J., Vries, D. de & Vogel, P. 1993, *Acoustic Control by Wave Field Synthesis*, Journal of the Acoustical Society of America, 93(5):2764-2778
- Kneppers, M., & Graaff, B. van der, SpaceJockey, <http://avdl1064.oli.tudelft.nl/WFS/>
- Rutz, H.H., Meloncillo, <http://www.sciss.de/meloncillo/index.html>
- Torger, A., BruteFIR, <http://www.ludd.luth.se/~torger/brutefir.html>
- Wright, M., Freed, A. & Momeni, A. 2003, “OpenSoundControl: State of the Art 2003”, *2003 International Conference on New Interfaces for Musical Expression*, McGill University, Montreal, Canada 22-24 May 2003, Proceedings, pp. 153-160